Introduction

Table of Contents

The USPS Handwritten Digit database is a set of handwriting data containing 1100 examples of the digits 0-9. To process and learn this data, a pocket algorithm was implemented that analyzes the intensity and symmetry of the input image.

Data preperation

To prepare for training, data is pulled from the USPS handwriting set and processed. `getfeatures()` is called to find the symmetry and intensity of each data point. The resulting set is then sliced to extract only the data for the numbers 1 and 5.

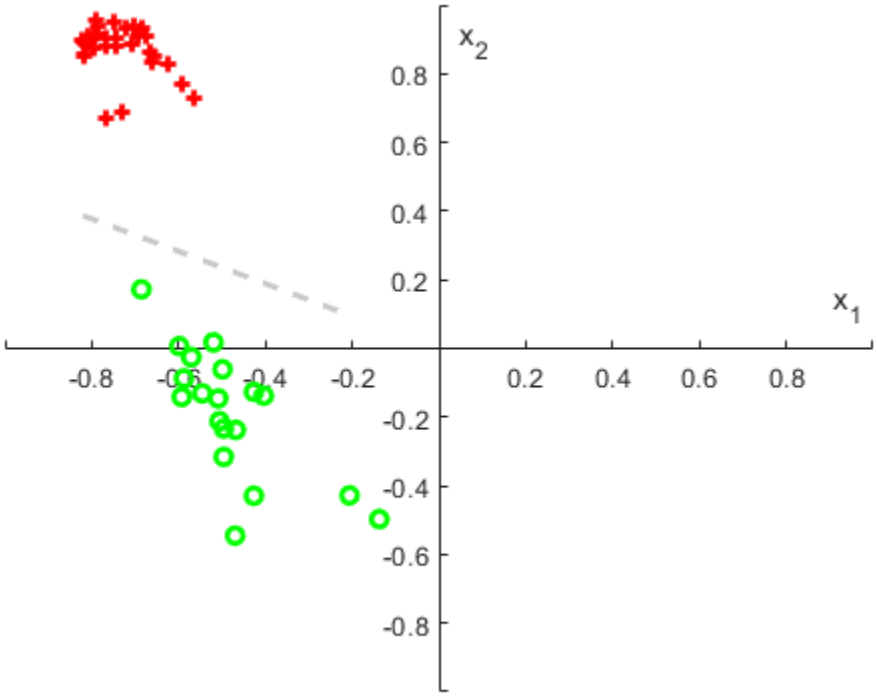
```
load('usps_modified.mat');
[features,classifications] = getfeatures(data);
% get the slices we want
xd = features([1:500 2001:2500],:);
yd = classifications([1:500 2001:2500]);
% 1's are classified as (1) and 5's are classified as (-1)
for i = 501:1000
    yd(i) = -1;
end
```

Analysis

When averaged over 1000 runs, The pocket algorithm performs admirably for both `N=50` and `N=200` with no changes needed. For `N=50`, the average in-group and out-group errors are 94.5% and 93.8%, respectively. For `N=200`, accuracy is 83.1% and 99.38%. Considering that the data set isn't perfectly linearly seperable, this accuracy is superb. To keep the algorithm running quickly, both the PLA and pocket algorithms are limited to 9 iterations. Anything beyond this brings diminishing returns. The in-group is randomly determined for each pocket, which should theoretically provide more guess variety and thus a better chance at a more accurate guess. No changes were necessary between the two sizes of N.

```
g = [0;1;0.5];
a = 0;
N = 50;
acc = zeros(1000,2);
for k = 1:1000
    g = [0;1;0.5];
    ao = 0;
    for i = 1:9
        [g,ao,ai] = pocket(xd, yd, g,a,N);
    end
    acc(k,:) = [ao;ai];
end
mao = mean(acc(:,1));
```

```
mai = mean(acc(:,2));
% plot an example
[x_t,y_t,x,y] = split_data(xd,yd,N);
[g_e,~] = pla(x_t, y_t);
pla_plot(x_t,y_t,g_e);
```



Published with MATLAB® R2021a